

Rassal Bölümlenmiş Veri Üzerinde Aşırı Öğrenme Makinesi ve Topluluk Algoritmaları ile Sınıflandırma

Classification with Extreme Learning Machine and Ensemble Algorithms Over Randomly Partitioned Data

Ferhat Özgür Çatak
Siber Güvenlik Enstitüsü
TÜBİTAK BİLGEM, Kocaeli, Türkiye
ozgur.catak@tubitak.gov.tr

Özetçe —Yaşadığımız Büyük Veri çağında, makine öğrenmesi tabanlı veri madenciliği yöntemleri, yüksek boyutlu veri setlerinin analiz edilmesinde yaygın olarak kullanılmaktadır. Bu tip veri setlerinden kullanışlı tahmin modellerinin çıkarılması işlemi, yüksek karmaşıklık nedeniyle zorlayıcı bir problemdir. Veri erişiminin yüksek seviyelere ulaşmasının sağladığı fırsatla, bunların otomatik olarak sınıflandırılması önemli ve karmaşık bir görev olmaya başlamıştır. Dolayısıyla, bu bildiride güvenilir sınıflandırma tahmin model kümelerinin oluşturulması için MapReduce tabanlı dağıtık Aşırı Öğrenme Makinesi (AÖM) araştırılmıştır. Buna göre, (i) veri kümesi toplulukları oluşturulması (ii) AÖM kullanılarak zayıf sınıflandırma modellerin oluşturulması ve (iii) zayıf sınıflandırma model kümesi ile güçlü sınıflandırma modeli oluşturulmuştur. Bu eğitim yöntemi, genel kullanıma açık bilgi keşfi ve veri madenciliği veri setlerine uygulanmıştır.

Anahtar Kelimeler—*Aşırı Öğrenme Makinesi, AdaBoost, Büyük Veri, Topluluk Metodları, MapReduce*

Abstract—In this age of Big Data, machine learning based data mining methods are extensively used to inspect large scale data sets. Deriving applicable predictive modeling from these type of data sets is a challenging obstacle because of their high complexity. Opportunity with high data availability levels, automated classification of data sets has become a critical and complicated function. In this paper, the power of applying MapReduce based Distributed AdaBoosting of Extreme Learning Machine (ELM) are explored to build reliable predictive bag of classification models. Thus, (i) dataset ensembles are build; (ii) ELM algorithm is used to build weak classification models; and (iii) build a strong classification model from a set of weak classification models. This training model is applied to the publicly available knowledge discovery and data mining datasets.

Keywords—*Extreme Learning Machine, AdaBoost, Big Data, Ensemble Methods, MapReduce*

I. GİRİŞ

Dünya genelinde bilgisayarlar, cep telefonları ve sensörler gibi cihazlar tarafından üretilen bilgi üzerinde hem büyüklük hem de çeşit bakımından oldukça yüksek miktarda

artış yaşanmaktadır. Bilgisayar teknolojisinin gelişmesiyle beraber büyük veri olarak adlandırdığımız konsept hemen her türde bilginin depolanmasına odaklanmıştır. Yüksek boyutlu veriden kullanılabilir tahmin modellerinin çıkarılması işlemi artık büyük veri kavramının içerisinde düşünülmektedir. Bu yüksek boyutlu verinin tahmin modellerinde kullanılmasının artmasıyla beraber, öğrenme algoritmalarının eğitiminin karmaşıklığıda artmaktadır. Bu nedenden dolayı, yüksek boyutlu veri setlerinin verimli bir şekilde işlenebilmesi için çeşitli topluluk metodları ve sınıflandırma algoritmalarını birleştiren makine öğrenmesi yöntemleri geliştirilmesi gerekmektedir.

Aşırı öğrenme makinesi (AÖM), Huang tarafından [1], genelleştirilmiş tek katmanlı ileri beslemeli ağ yapısı temel alınarak geliştirilmiştir. AÖM, düşük eğitim zamanı, çok sınıflı eğitim kümelerinde yeni örnekler üzerinde yüksek genelleme özelliği ve herhangi bir eğitim parametresi içermemesi gibi avantajlarından dolayı, doküman sınıflandırma [2], biyoenformatik [3], görüntü tanıma [4] gibi bir çok farklı alanda kullanılmaktadır.

Son yıllarda araştırmacılar, tahmin modellemesi için dağıtık ve paralel çatılarla ilgili yöntemler geliştirmektedirler. Çalışmaların çok az bir kısmı MapReduce yöntemini kullanmaktadır. Bu çalışmada önerilen yöntem, yüksek boyutlu veri setlerinden tahmin modeli oluşturmak için, farklı boyutlarda rassal veri parçaları oluşturularak bunları eğitim aşamasında kullanmak, bu şekilde AÖM algoritması ve AdaBoost yöntemi ile sınıflandırma fonksiyon kümesi oluşturmaktadır. MapReduce kullanılarak, veri setinden alt veri parçaları oluşturularak eğitilen AdaBoost, topluluk yöntemleri ile birleştirilerek, tekil bir global sınıflandırma fonksiyonu ortaya çıkarılmaktadır. Çalışmanın en önemli katkıları şu şekildedir:

- Genelleştirilmiş MapReduce tekniği temelli AdaBoost AÖM sınıflandırma modeli ile daha hızlı ve daha iyi sınıflandırma performansına sahip model elde edilmektedir.
- Bu çalışmanın önerdiği yeni öğrenme yöntemi ile elde edilen paralel eğitim, yüksek boyutlu veri setlerinin öğrenme için kullandığı hesaplama zamanını azaltmaktadır.

- Eğitim esnasında kullanılan her bir düğüm (node) diğerinden bağımsız olmasından dolayı veri haberleşmesi azalmaktadır.

II. ÖN BİLGİLER

Bu bölümde, çalışmada kullanılan AÖM, AdaBoost ve MapReduce hakkında bilgi verilecektir.

A. Aşırı Öğrenme Makinesi

AÖM, ilk olarak tek katmanlı ileri beslemeli sinir ağı olarak geliştirilmiştir [1]. Daha sonra yapılan çalışmalarda gizli katmanın sadece nöron olmadığı genelleştirilmiş tek katmanlı ileri beslemeli ağ önerilmiştir [7]. AÖM, oluşturduğu sinir ağının giriş ağırlıkları ile gizli düğüm eğimi değerlerini rassal olarak oluşturmakta ve çıktı katmanını ağırlıkları en küçük kareler yöntemi ile hesaplamaktadır [8].

Bilinmeyen bir \mathcal{X} dağılımından elde edilen bağımsız özdeşçe dağılmış eğitim veri kümesi $\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid i = \{1, \dots, n\}, \mathbf{x}_i \in \mathbf{R}^p, y_i \in \{1, 2, \dots, K\}\}$ olsun. Sinir ağının hedefi $f : \mathcal{X} \rightarrow \mathcal{Y}$ şeklinde fonksiyonu bulmaktır. N gizli düğüme sahip tek katmanlı ileri beslemeli sinir ağı Denklem 1'de tanımlanmıştır.

$$f_N(\mathbf{x}) = \sum_{i=1}^N \beta_i G(\mathbf{a}_i, b_i, \mathbf{x}), \mathbf{x} \in \mathbf{R}^n, \mathbf{a}_i \in \mathbf{R}^n \quad (1)$$

\mathbf{a}_i ve b_i öğrenme parametresi, β_i ise i . gizli düğümün ağırlığıdır. Genelleştirilmiş tek katman ileri besleme sinir ağı için AÖM'nin çıktı fonksiyonu Denklem 2'de gösterilmiştir.

$$f_N(\mathbf{x}) = \sum_{i=1}^N \beta_i G(\mathbf{a}_i, b_i, \mathbf{x}) = \beta \times h(\mathbf{x}) \quad (2)$$

İkili sınıflandırma uygulamaları için AÖM karar fonksiyonu ise Denklem 3'de gösterilmiştir.

$$f_N(\mathbf{x}) = \text{sign} \left(\sum_{i=1}^N \beta_i G(\mathbf{a}_i, b_i, \mathbf{x}) \right) = \text{sign}(\beta \times h(\mathbf{x})) \quad (3)$$

Denklem 2 diğer bir form olarak Denklem 4'de gösterilmiştir.

$$H\beta = T \quad (4)$$

H ve T sırasıyla gizli katman matrisi ve çıktı matrisidir. Gizli katman matrisi Denklem 5'de gösterilmiştir.

$$H(\tilde{\mathbf{a}}, \tilde{\mathbf{b}}, \tilde{\mathbf{x}}) = \begin{bmatrix} G(a_1, b_1, x_1) & \cdots & G(a_l, b_l, x_1) \\ \vdots & \ddots & \vdots \\ G(a_1, b_1, x_N) & \cdots & G(a_l, b_l, x_N) \end{bmatrix}_{N \times L} \quad (5)$$

burada $\tilde{\mathbf{a}} = a_1, \dots, a_L$, $\tilde{\mathbf{b}} = b_1, \dots, b_L$, $\tilde{\mathbf{x}} = x_1, \dots, x_N$ şeklindedir. Çıktı matrisi Denklem 6'de gösterilmiştir.

$$T = [t_1 \dots t_N] \quad (6)$$

B. AdaBoost

AdaBoost, nitelik matrisi X ve çıktı sınıfları, $y \in \{+1, -1\}$, kullanarak, zayıf öğrenme modelleri, $h_t(\mathbf{x})$, birleştirerek $H(\mathbf{x})$ şeklinde güçlü sınıflandırma modeli oluşturmaya çalışmaktadır [9]. Denklem 7'de güçlü sınıflandırma modeli gösterilmektedir.

$$H(\mathbf{x}) = \text{sign}(f(\mathbf{x})) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(\mathbf{x}) \right) \quad (7)$$

Tablo I: Sık kullanılan değişkenler ve notasyonlar.

Notasyon	Açıklama	Notasyon	Açıklama
M	Veri parça bölümlene uzunluğu	T	AdaBoost T boyutu
h	Sınıflandırma fonksiyonu	nh	AÖM'de kullanılan gizli düğüm sayısı
X_m	Veri seti, \mathcal{D} , girdi değerlerinin m veri parçası	Doğ	Sınıflandırma hipotezinin doğruluğu
Y_m	Veri seti, \mathcal{D} , çıktı değerlerinin m veri parçası	H.	Hassasiyet
ϵ	Hata oranı	G.C.	Geri Çağırım

C. MapReduce

MapReduce yöntemi yüksek boyutlu veri setlerinin işlenmesine olanak sağlayan, ayrıca Google tarafından da oldukça sık kullanılan bir programlama modelidir [10]. Kullanıcılar tarafından tanımlanan *Map* ve *Reduce* fonksiyonları ve bu fonksiyonlara girdi değeri olarak verilen anahtar/değer dizileri (Key/Value pairs) kullanılmaktadır. MapReduce yönteminde kullanılan anahtar değer veri modeli genellikle ilişkisel veri modelleri ile tasarlanamayacak veri setlerine uygulanmaktadır. Örnek olarak bir web sayfasının adresi anahtar değerine yazılırken bu sayfanın HTML içeriği ise değer alanına yazılmaktadır. Grafik tabanlı veri modellerinde ise anahtar alanı düğüm anahtar (id) bilgisini içerirken değer ise liste olarak kendisine komşu olan düğümlerin anahtar bilgilerini içerebilir.

Map fonksiyonu paralel olarak girdi veri setinde bulunan her ikiliye uygulanmaktadır. Fonksiyon bir veri alanında bulunan veri çiftlerini alarak bunları farklı bir alana veri çift listesi olarak vermektedir.

$$\text{Map}(a_1, d_1) \rightarrow \text{liste}(a_2, d_2) \quad (8)$$

Reduce fonksiyonu ise yine paralel olarak *Map* fonksiyonu tarafından ilişkilendirilmiş anahtar değer yapısına uygulayarak yeni değerler listesi oluşturmaktadır.

$$\text{Reduce}(a_2, \text{liste}(d_2)) \rightarrow \text{liste}(a_3, d_3) \quad (9)$$

MapReduce çatısının anahtar/değer şeklindeki çiftlerden oluşan listeyi değerler listesi şekline çevirmektedir.

III. SİSTEM MODELİ

Bu bölümde, MapReduce temelli AdaBoost AÖM algoritmasının detayları verilecektir. Temel fikir Bölüm III-A kısmında anlatılacaktır. Sistemin gerçekleştirimi ise Bölüm III-B kısmında tanımlanacaktır.

A. Temel Fikir

AdaBoost temelli AÖM sınıflandırma algoritmasının hesaplanması aşamasının dağıtık ve paralel hale getirilmesi bu çalışmanın esas görevidir. Önerilen yöntemin temel fikri, sınıflandırma topluluk fonksiyonlarının rassal veri parçaları (X_m, Y_m) kullanılarak paralel olarak hesaplanmasıdır.

Tablo I'de, bildirinin anlaşılmasında kolaylık olması için çalışmada kullanılan değişken ve notasyonların özeti verilmiştir.

B. Modelin Gerçekleştirimi

MapReduce temelli AdaBoost AÖM algoritmasının sözde kodu Algoritma 1 ve Algoritma 2’de gösterilmiştir. Önerilen öğrenme modelinin *Map* metodu, bölümlene boyutu, M , aralığına kadar tam sayı olacak şekilde rassal değerlerin eğitim veri kümesinin her bir satırına atanması şeklindedir. *Map*’in girdi değeri olan \mathbf{x} , eğitim veri kümesi \mathcal{D} ’nin bir satırıdır. *Map* metodu girdi matrisini satır olarak bölümlenmekte ve $\langle \text{rassalBolumId}, \mathbf{x} \rangle$ anahtar/değer ikililerini oluşturmaktadır. *rassalBolumId*, veri parçasının tanımlayıcısı olarak atanmakta ve anahtar olarak *Reduce* aşamasına transfer edilmektedir. *Reduce* aşamasının sözde kodu Algoritma 2’de

Algorithm 1 AdaBoostAÖM::Map

- 1: **Girdi:**
 $(\mathbf{x}, y) \in \mathcal{D}, M$
 - 2: $k \leftarrow \text{rand}(0, M)$
 - 3: *Output*($k, (\mathbf{x}, y)$)
-

gösterilmiştir. *Reduce* aşaması, Algoritma 2 sözde kodunun 3 – 8. satırları arasında bulunan döngüde gerçekleştirilmiştir. Her bir veri parçası, $(\mathbf{X}_k, \mathbf{y}_k)$, AdaBoost topluluk yöntemi temelli AÖM ile eğitilmektedir. Böylece her bir *Reduce* işlemi ayrı bir sınıflandırma modeli ortaya çıkarmaktadır. *Reduce* işleminde anahtar, k , *Map* aşamasında rassal olarak atanan *rassalBolumId*, girdi olarak kullanılmaktadır.

Algorithm 2 AdaBoostAÖM::Reduce

- 1: **Girdi:**
 Anahtar k , Deger V , T
 - 2: $(\mathbf{X}_n, \mathbf{y}_n) \leftarrow V$
 - 3: **for** $t = 1..T$ **do**
 - 4: $h_t \leftarrow AOM(\mathbf{X}_n, \mathbf{y}_n)$
 - 5: $\mathbf{y}_{pred}, \epsilon_t \leftarrow h_t(\mathbf{X}_n)$
 - 6: $\alpha_t \leftarrow \frac{1}{2} \ln \frac{1 - \epsilon_t}{\epsilon_t}$
 - 7: $\mathcal{D}_{t+1} = \frac{\mathcal{D}_t \times \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$
 - 8: **end for**
 - 9: **Çıktı:**
 $h_m = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(\mathbf{x}) \right)$
-

IV. BENZETİM SONUÇLARI

Bu bölümde, internet ortamında açık olarak erişilebilen gerçek veri setleri kullanılarak, önerilen modelin sınıflandırma performansı farklı ölçüm yöntemleri ile sınanmıştır. Gerçekleştirim aşamasında 64 bit Python 2.7 yazılım dili ve MrJob kütüphanesi kullanılmıştır.

Bölüm IV-A’da, deneysel ortamda kullanılan veri setleri ve AÖM’nin parametreleri açıklanmaktadır. Standart AÖM’nin her bir veri setinde sınıflandırma performanslarının saklı düğümlere göre değişimi Bölüm IV-B’de gösterilmektedir. Bölüm IV-C’de, önerilen öğrenme modelinin deneysel sonuçları tablo ve grafik olarak gösterilmektedir.

A. Deneysel Kurulum

Bu bölümde, önerilen yöntem, Pendigit, Letter, Statlog ve Page-blocks şeklinde dört farklı veri seti kullanılarak sınıflandırma modeli oluşturulmuş, bu şekilde yöntemin etkinliği ve verimliliği ölçümlenmeye çalışılmıştır. Kullanılan açık dört farklı veri kümesi Tablo II’de gösterilmektedir. Kullanılan bütün veri setleri, ikiden fazla etikete sahip, çok sınıflıdır.

Tablo II: Kullanılan veri setlerinin bilgileri.

Veri seti	# Eğitim	# Test	# Sınıf	# Öz nitelik
Pendigit	7495	3498	10	64
Skin	220543	24507	2	4
Statlog	43500	25000	10	7
Page-blocks	4500	973	5	10

B. Veri setlerinin Standart AÖM ile sonuçları

Tablo III’de çalışmanın deneysel kısmında kullanılan veri setlerinin AÖM sonuçları paylaşılmıştır. nh değeri, 1 – 500 arasında değişmektedir. Performans ölçümleri için doğruluk, hassasiyet, geri çekilme ve F_1 değerleri kullanılmıştır.

Tablo III: Veri setlerinin standart AÖM sonuçları.

Veri S.	nh .	Doğ.	H.	G.C.	F_1
Pendigit	149	0,8404	0,8393	0,8416	0,8407
Skin	98	0,9754	0,9956	0,9583	0,9894
Statlog	249	0,8871	0,8556	0,9237	0,9757
Page Blocks	498	0,9873	0,9794	0,9988	0,9977

C. Sonuçlar

Bu çalışma kapsamında kullanılan veri setlerinin sınıf dağılımları dengesiz olmasından dolayı optimal sınıflandırıcı hipotezin bulunmasında sadece geleneksel doğruluk tabanlı performans ölçümü yeterli değildir. Bu nedenle sınıflandırıcı hipotez ölçümünde ortalama doğruluk, ortalama hassasiyet, ortalama geri çekim [5], F_1 ölçümü şeklinde dört farklı yöntem kullanılmıştır. Kullanılan yöntemler bilgi çıkarımı alanında en çok kullanılan yöntemlerdir [6].

Hassasiyet, elde edilen ilgili örneklerin toplam elde edilen örneklerle oranıdır. Hassasiyet Denklem 10’da gösterilmektedir.

$$\text{Hassasiyet} = \frac{\text{Dogru}}{\text{Dogru} + \text{Hata}} \quad (10)$$

Geri çekilme, elde edilen ilgili örneklerin toplam ilgili örneklerle oranıdır. Geri çekilme Denklem 11’de gösterilmektedir.

$$\text{Geri Cekilme} = \frac{\text{Dogru}}{\text{Dogru} + \text{Kayip}} \quad (11)$$

Önerilen modelde, her bir sınıf için farklı olarak hassasiyet ve geri çekilme değerleri hesaplanıp toplam sınıf sayısına bölünerek, elde edilen sınıflandırma hipotezinin ortalama ölçüm değerleri hesaplanmaktadır. Ortalama hassasiyet ve geri çekilme Denklem 12 ve Denklem 13’de gösterilmektedir.

$$\text{Hassasiyet}_{ort} = \frac{1}{n_{sinif}} \sum_{i=0}^{n_{sinif}-1} \text{Hassasiyet}_i \quad (12)$$

$$\text{Geri Cekilme}_{ort} = \frac{1}{n_{sinif}} \sum_{i=0}^{n_{sinif}-1} \text{GeriCekilme}_i \quad (13)$$

F_1 ölçümü, hassasiyet ve geri çekilmenin harmonik ortalamasıdır. Değerlendirme modeli, Denklem 14’de gösterilen F_1

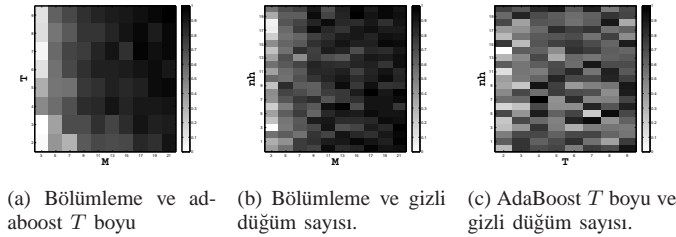
Tablo IV: Veri setlerinin en iyi performans sonuçları.

Veri S.	# C.	T	# H.N.	Doğ.	H.	G.C.	F_1
Pendigit	20	10	21	0,8256	0,8369	0,8234	0,8301
Skin	21	5	21	0,9892	0,9773	0,9913	0,9842
Statlog	11	2	21	0,9103	0,7486	0,5069	0,6045
Page Blocks	1	1	340	0,9404	0,9027	0,5756	0,7030

ölçümünün çok sınıflı halini kullanmaktadır.

$$F_1 = 2 \times \frac{Hassasiyet_{ort} \times GeriCekilme_{ort}}{Hassasiyet_{ort} + GeriCekilme_{ort}} \quad (14)$$

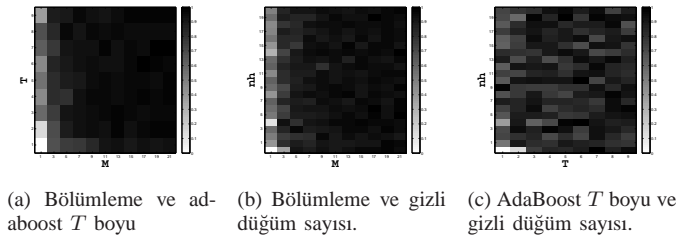
Yapılan ölçümlerin sonuçları Tablo IV'de gösterilmiştir. Her bir veri seti için bölümlleme, M , saklı düğüm sayısı, nh , AdaBoost iterasyon sayısı, T , parametrelerine göre doğruluk değişiminin sonuçları Şekil 1 – 4'de gösterilmiştir. Isı haritalarında renk, siyaha yaklaşması durumunda modelin doğruluğu artmaktadır. Ölçek her bir grafiğin yanında bulunan renk çubuğu ile verilmiştir.



Şekil 1: Statlog veri kümesi ısı haritası.



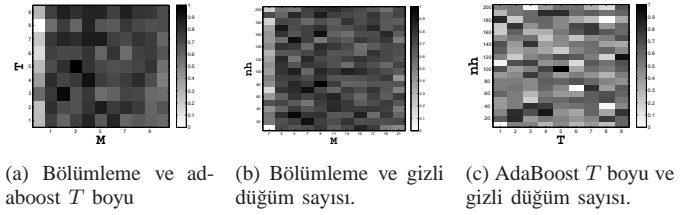
Şekil 2: Pendigit veri kümesi ısı haritası.



Şekil 3: Skin veri kümesi ısı haritası.

V. SONUÇ

Bu çalışmada, MapReduce temelli AdaBoost AÖM algoritması uygulaması, yüksek boyutlu veri setlerinin eğitilmesi için önerilmiştir. Girdi matrisini parçalara ayırarak, önerilen yöntem, AÖM sınıflandırmasının eğitim aşamasının karmaşıklığını azaltmaktadır. Matrisin parçalanması ile yaşanacak olan sınıflandırma performans azalmasının üstesinden gelmek için



Şekil 4: Page blocks veri kümesi ısı haritası.

AdaBoost yöntemi ile desteklenmiştir. Deneysel çalışmalarla elde edilen sonuçlarla, sadece yüksek boyutlu veri setlerinin eğitim karmaşıklığı azalmamakta ayrıca geleneksel AÖM algoritmasının sınıflandırma performansına göre artış yaşanmaktadır.

Bu çalışma kapsamında önerilen dağıtık AÖM yöntemi veri parça bölümlleme uzunluğu, M , AdaBoost yöntemi iterasyon sayısı, T , gizli katman düğüm sayısı, nh , şeklinde üç parametreye sahiptir. Isı haritası grafiklerinde gösterilen sonuçlara bakarak M ve T 'nin, nh parametresine göre sınıflandırma performans ölçümüne olan etkisi daha fazla olduğu gözlemlenmektedir.

Önerilen yöntem, yüksek boyutlu veri setlerinin karmaşıklığını, matrisi alt parçalara ayırarak, AÖM eğitim aşamasının zorluğunu azaltmaktadır. Tablo III ve Tablo IV karşılaştırıldığında, model karmaşıklık göstergesi olarak düşünülen, nh sayısında azalma olduğu görülmektedir. Bu nedenle, yöntem sadece girdi matrisi karmaşıklığını değil aynı zamanda model karmaşıklığında azalmasını sağlamaktadır.

KAYNAKÇA

- [1] Guang-Bin Huang, Qin-Yu Zhu, Chee-Kheong Siew, "Extreme learning machine: Theory and applications", *Neurocomputing*, Volume 70, Issues 1–3, Pages 489-501, December 2006
- [2] Xiang-guo Zhao, Guoren Wang, Xin Bi, Peizhen Gong, Yuhai Zhao, XML document classification based on ELM, *Neurocomputing*, Volume 74, Issue 16, Pages 2444-2451, September 2011
- [3] Guoren Wang, Yi Zhao, Di Wang, A protein secondary structure prediction framework based on the Extreme Learning Machine, *Neurocomputing*, Volume 72, Issues 1–3, Pages 262-268, December 2008
- [4] Lan, Yuan, Hu, Zongjiang, Soh, Yeng Chai and Huang, Guang-Bin. "An extreme learning machine approach for speaker recognition", *Neural Computing and Applications*, Volume 22, no. 3-4, 417-425, 2013
- [5] Andrew Turpin; Falk Scholer. "User performance versus precision measures for simple search tasks", *In Proceedings of the SIGIR '06*, ACM, New York, NY, USA, 2006
- [6] Makhoul, J.; Kubala, F.; Schwartz, R.; Weischedel, R., "Performance Measures For Information Extraction", *Proceedings of the DARPA Broadcast News Workshop*, 1999
- [7] Guang-Bin Huang, Lei Chen, Convex incremental extreme learning machine, *Neurocomputing*, Volume 70, Issues 16–18, Pages 3056-3062, October 2007
- [8] Jiexiong Tang; Chenwei Deng; Guang-Bin Huang; Baojun Zhao, "Compressed-Domain Ship Detection on Spaceborne Optical Image Using Deep Neural Network and Extreme Learning Machine", *IEEE Transactions on Geoscience and Remote Sensing*, vol.53, no.3, pp.1174,1185, 2015
- [9] Iago Landesa-Vázquez, José Luis Alba-Castro, Double-base asymmetric AdaBoost, *Neurocomputing*, Volume 118, Pages 101-114, 2013
- [10] Dean, Jeffrey, and Sanjay Ghemawat. "MapReduce: simplified data processing on large clusters.", *Communications of the ACM*, vol. 51, no.1 (2008), 107-113,2008

